National Defence
Défense nationale

DATA SELECTION FOR FAST PROJECTION TECHNIQUES APPLIED TO ADAPTIVE NULLING: A COMPARATIVE STUDY OF PERFORMANCE (U)

by

Mylène Toulgoat and Ross M. Turner

DEFENCE RESEARCH ESTABLISHMENT OTTAWA
REPORT NO. 1100

Canada

December 1991
Ottawa

National Defence / Défense nationale

DATA SELECTION FOR FAST PROJECTION TECHNIQUES APPLIED TO ADAPTIVE NULLING: A COMPARATIVE STUDY OF PERFORMANCE (U)

by

Mylène Toulgoat and Ross M. Turner
Surface Radar Section
Radar Division

DEFENCE RESEARCH ESTABLISHMENT OTTAWA
REPORT NO. 1100

PCN
041LC

December 1991
Ottawa

Data Selection for Fast Projection Techniques Applied to Adaptive Nulling: A Comparative Study of Performance

by

Mylène Toulgoat and Ross M. Turner

ABSTRACT

This report describes simulation studies of fast projection techniques for adaptive nulling of jammer signals. These techniques are based on a data selection criterion applied directly to the data vectors obtained at the output of an antenna array. The selected data vectors are used in fast projection algorithms to estimate the antenna weighting vector that is orthogonal to the jamming vectors. Four such algorithms are described ranging from the fastest but least effective to the most computationally demanding but most effective. In effect, these algorithms provide a better trade–off of performance versus computational load than has heretofore been available. Relative performance is compared with that of the standard Sample Matrix Inversion (SMI) technique for test cases which evaluate the effects of the number of jammers, the jammer strengths relative to the receiver noise and the jammer angular positions.

Résumé

Ce document est une étude, basée sur des simulations, de la performance relative des techniques de projection rapides pour la formation auto–adaptive des faisceaux. Ces algorithmes utilisent un critère pour la sélection de bons vecteurs de données à la sortie d'une antenne réseau. On utilise les vecteurs selectionnés dans les algorithmes de projection rapides afin d'estimer le vecteur de pondération qui est orthogonal aux brouilleurs. On décrit quatre algorithmes, le premier étant le plus performant du point de vue du nombre de calculs mais le moins performant du point de vue d'annulation des brouilleurs, le dernier étant moins rapide mais donnant une meilleure annulation. En effet, ces algorithmes donnent un meilleur compromis entre le rendement et la charge de calculs que les algorithmes connus jusqu'à présent. On a comparé le rendement des quatres techniques avec celui de la technique "sample matrix inversion ou SMI" pour trois scénarios différents; ces trois derniers démontrent les effets du nombre de brouilleurs, de leur position, et de leur puissance par rapport au bruit du fond.

Accession For	
NTIS GRA&I	☑
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

EXECUTIVE SUMMARY

Jamming of radar systems via the antenna sidelobes or the antenna mainbeam can seriously degrade radar performance. Thus jamming can provide a crucial advantage to enemy forces if radar electronic–counter–counter–measures (ECCM) are not effective.

Adaptive antenna nulling provides an effective ECCM but at the cost of increased equipment complexity and capability. A major cost factor is the need for a very high speed, real–time, computational capability. The computational capability required is so high it limits the application of this effective ECCM.

This report describes and evaluates computationally efficient algorithms for adaptive nulling; these reduce the requirements for high speed computation and hence the cost of implementation.

The algorithms to be described and evaluated are based on projection techniques; the jammer vector space is estimated from the data vectors obtained from the antenna array elements. An antenna weight vector is constructed by projecting the steering weight vector into a space that is orthogonal to the estimated jamming vector space. When the adapted weight vector is applied to superimposed signal and jamming, the jamming signals are greatly reduced while the loss of signal is small. This occurs because the adapted weight vector is approximately orthogonal to the jamming signals.

Four versions of the algorithm are described and evaluated, all based on data selection criteria followed by Gram–Schmidt orthogonalization. The algorithms are named as follows: (1) DVO for data vector orthogonalization, (2) DVSO for data vector selection and orthogonalization, (3) DVSO–WAVER for repeated applications of DVSO followed by averaging of the weight vectors found in each application of DVSO, (4) DVSO–COVAR for repeated applications of DVSO followed by a combining of the selected data vectors into a covariance matrix which in turn is is followed by a final application of DVSO to the columns of the covariance matrix.

The above four algorithms are evaluated by means of simulations and compared with the Sampled Matrix Inversion (SMI) technique. The latter, the standard method for finding the adapted weight vector, is very computationally demanding. In terms of performance versus computational load, all four techniques compare favorably with the SMI technique. The four techniques trade off performance against computational load, DVO being the fastest in terms of computations but providing the least jammer cancellation while DVSO–COVAR provides the best cancellation performance but at the cost of a considerable computational load.

CONTENTS

Data Selection for Fast Projection Techniques Applied to Adaptive Nulling: A Comparative Study of Performance

1.0 INTRODUCTION

Modern signal–processing algorithms achieve effective adaptive beam forming – but at a cost of a very high computational load. This computational load can be prohibitive when the data rate is high i.e., when the weight vector must be changed frequently in a rather short period of time as in many radar applications. The well known Sample Matrix Inversion (SMI) method [1] provides an estimate of the weight vector that maximizes the signal to interference ratio. Since the receiver noise is averaged when many samples are used in the formation of the covariance matrix, the error decreases with the number of samples used. Unfortunately, the calculation and inversion of the sample covariance matrix are very computationally intensive, particularly for large antenna arrays.

Projection methods based on eigenvector analysis, such as the MUSIC technique [2], are effective for both interference suppression and spectral estimation. These techniques yield estimates of the noise and signal subspace with an error that decreases with the number of sample vectors used in the covariance matrix. These methods exploit averaging by incorporating many data vectors into an estimated covariance matrix which approaches the true covariance matrix asymptotically as the observation period increase. Such an eigenanalysis is computationally demanding. Projection methods, however, need not be based on eigenanalysis; large decreases in the computational load can be achieved by faster projection methods at the expense of some performance.

A fast projection method based on Gram–Schmidt orthogonalization [3,4,5] provides significant computational savings compared with the SMI, Capon or MUSIC techniques. This method employs direct orthogonalization of the complex data vectors obtained from a sampled–aperture array to form a set of basis vectors describing the interference vector space. This basis is used to construct an adapted weight which is, ideally, orthogonal to the interference subspace and hence orthogonal to all jammer signals. It is essential that the signal be absent when the data vectors are sampled in order to avoid cancellation of the desired signal. This is a standard radar problem; one solution is to sample at certain time intervals corresponding to ranges where it is known that there are no targets or clutter returns.

In practice, the interference subspace is estimated with a small number of data vectors corrupted with receiver noise, the estimate is not perfect and the resulting Signal–to–Noise–plus–Jammer Ratio (SNJR) is reduced accordingly. The estimation of the jammer subspace and the adapted weight vector require the order of N^2K complex multiplications as opposed to K^3 in the SMI case. Here K is the number of array elements and N is the number of data vectors used for the orthogonalization. When $N << K$, $N^2K << K^3$ and the technique is considerably more computationally efficient than the SMI method. We use the name Data Vector Orthogonalization (DVO) to describe this process and to differentiate it from numerous other implementations of the Gram–Schmidt process.

A modified version of the DVO algorithm uses a data selection method prior to Gram–Schmidt (GS) orthogonalization in order to choose the "best" M data vectors out of a larger set, N, to provide an enhanced SNJR. The new

approach, called Data Vector Selection and Orthogonalization (DVSO) gives performance superior to DVO in terms of achievable SNJR for a given number of complex multiplications [6].

Neither DVO nor DVSO benefit from averaging in the same way as do the MUSIC and SMI methods. In the MUSIC and SMI methods, averaging is a major contribution to the computational load – the other major contribution coming from the determination of the weight vector which requires inverting the covariance matrix for the SMI method or the calculation of the eigenvectors for the MUSIC method.

We have developed two new algorithms [7] that incorporate averaging techniques into the DVSO method to provide an enhanced SNJR. The first, called DVSO–WAVER, averages repeated computations of the adapted weight vectors. The second, called DVSO–COVAR, uses a data selection process for the choice of the data vectors used in the formation of the covariance matrix. Gram–Schmidt orthogonalization is then applied to the columns of the covariance matrix. These two methods provide essentially the same SNJR as the SMI method but require far fewer computations.

This report is an exhaustive simulation study of the relative performance of fast projection techniques for adaptive nulling: DVO, DVSO, DVSO–WAVER and DVSO–COVAR. In order to simplify the comparison, we have defined three test cases designed to examine the effects of various jammer characteristics on the relative performance of the techniques. Following the introduction, we describe the signal and jammer characteristics, the simulation, the test cases and the basis of comparison. We then describe each technique in turn and its performance with respect to the test cases. Finally, we carry out a comparison of the techniques.

2.0 SIGNAL AND JAMMER CHARACTERISTICS

2.1 SIGNAL CHARACTERISTICS

Radars commonly produce pulsed signals which have narrow bandwidths with respect to the carrier frequency. Targets of interest are in the far field so that the received signals arrive as plane waves. All evaluations are carried out for an array steered in the broadside direction; the desired signal is therefore defined as a plane wave arriving from the broadside direction. The results are applicable to other steering directions provided account is taken of the reduced antenna gain when the array is steered away from broadside.

2.2 JAMMER CHARACTERISTICS

Radar systems are usually narrow band; this means that the received jammer wave front has nearly perfect spatial correlation over the face of the array provided the time taken to cross it is small compared with the reciprocal of the system bandwidth. Provided no radar reflections are received at the time of sampling, the n^{th} data vector at time t_n is expressed as

$$\mathbf{x}_n = \sum_{i=1}^{L} j_i(t_n)\ \mathbf{a}_i + \mathbf{n}_n$$

where L is the number of jammers, $\mathbf{n}_n$ is a receiver–noise vector with mutually independent components and power, $E|\mathbf{n}_n|^2 = K\,\sigma^2$, with K the number of array elements. The quantity $j_i(t_n)$ is a complex gaussian random variable representing the i^{th} jammer amplitude at time t_n, and $\mathbf{a}_i$ is a deterministic vector representing the direction of arrival of the i^{th} jammer defined as

$$\mathbf{a}_i^T = [1,\ \exp(j2\pi d\sin\theta_i/\lambda),\ldots,\ \exp(j2\pi d(K-1)\sin\theta_i/\lambda)]$$

Here d is the inter–element spacing, and θ_i is the direction of arrival of the i^{th} jammer. The sample vectors or "snapshots" are taken at such intervals that $j_i(t_n)$ and $j_i(t_{n-1})$ are completely independent.

3.0 SIMULATIONS AND PERFORMANCE MEASURES

3.1 MONTE CARLO SIMULATIONS

Monte Carlo simulations are used to evaluate the performance of the various techniques. One hundred independent trials are carried out for each parameter setting under test with performance measures averaged over these trials. Jammers are simulated as a zero–mean complex gaussian process having perfect spatial correlation over the array and zero temporal correlation from one data vector to the next. This gives rise to a jammer amplitude that is Rayleigh distributed. Receiver noise is also simulated as a zero–mean complex gaussian random process but having zero spatial correlation over the array and zero temporal correlation from one data vector to the next. The receiver noise power at the element level is set at $\sigma^2 = 10^{-4}$. The signal is simulated as a plane wave arriving from the broadside direction with an amplitude such that the signal–to–noise–ratio, (SNR), is zero dB at the element output. This allows convenient scaling of the results for different SNRs: simply add the actual SNR in dB to the final Signal–to–Noise–plus–Jammer–Ratio (SNJR) determined from the simulation.

Almost all of the results are presented for a ten–element array of omnidirectional elements with an element spacing of one half wavelength. In the last section, however, we present results for a forty–element array in order to demonstrate the effects of using a larger array. The smaller ten–element array was used for the majority of the results to keep the computational load to a reasonable level.

3.2 PERFORMANCE MEASURES

The ultimate measure of performance of an adaptive array is the Signal–to–Noise–plus–Jammer Ratio (SNJR). We attempt to maximize this quantity by using an adapted weight vector, $\mathbf{w}_a$, that discriminates against jamming while causing a minimal loss of useful signal. A standard of comparison is the optimal SNJR obtained when the adapted weight vector is computed using perfect knowledge of the interference and noise covariance matrices. While the true covariance matrix can be calculated in a simulation, it can only be estimated in an operational situation. Another important measure of performance is the Improvement Factor which is the SNJR (dB) achieved with adaptive weighting minus the SNJR (dB) achieved with uniform antenna weighting. A final measure of performance is the difference in dB between the optimal SNJR and the achieved SNJR. These performance measures will be averaged over the 100 Monte–Carlo trials at each parameter setting and plotted versus the number of complex multiplications or against other quantities or parameters that will be

defined for the individual techniques when these are described.

3.3 TEST CASES

We have specified three test cases, T_1, T_2 and T_3 which are designed to test the effects of the number of jammers, the Jammer–to–Noise Ratio, JNR, and the jammer positions.

3.3.1 TEST CASE T_1

Here we have specified the Jammer–to–Noise Ratio, JNR, at the receiving element output to be 40 dB. The number of jammers is designated as L. For L=1, the jammer is at an angle of $\theta_1 = -17°$. For L=3, we have three jammers at $\theta_1 = -17°$, $\theta_2 = 30°$ and $\theta_3 = 65°$. For L=5, the jammers are at $\theta_1 = -30°$, $\theta_2 = -17°$, $\theta_3 = 30°$, $\theta_4 = 50°$ and $\theta_5 = 65°$.

3.3.2 TEST CASE T_2

Here we have defined three equi–powered jammers at $\theta_1 = -17°$, $\theta_2 = 30°$ and $\theta_3 = 65°$. We consider four values for JNR of 10, 20, 30 and 40 dB as measured at the element level.

3.3.3 TEST CASE T_3

In this case, there are three jammers. We have defined four scenarios: D_1, D_2, D_3 and D_4.

	θ_1	θ_2	θ_3
D_1	–20°	10°	35°
D_2	20°	40°	60°
D_3	–3°	–30°	–45°
D_4	–17°	30°	65°

All jammers have JNR=40dB. Since a 10–element array with halfwave length spacing has a beamwidth of about 13° and the array is steered to broadside, D_3 is a mainbeam jamming situation because of the jammer at –3°. Since the achieved SNJR is much less for mainbeam jamming than for sidelobe jamming, the SNJR results for D_3 are plotted separately from the results for the other scenarios.

4.0 DATA VECTOR ORTHOGONALIZATION

A phased array is steered by means of a weight vector, $\mathbf{w}_s$, that provides the phase weighting required to focus the array in the desired direction of search. Projection techniques achieve jammer cancellation by projecting $\mathbf{w}_s$ into a space orthogonal to the jamming subspace; the resulting projection, the adapted weight vector, $\mathbf{w}_a$, is orthogonal to the jamming by construction. Provided the jamming signals are well removed from the mainbeam direction, i.e. in the antenna sidelobes, the mainbeam gain is only slightly reduced when $\mathbf{w}_a$ is used instead of $\mathbf{w}_s$.

Data Vector Orthogonalization (DVO) is an efficient method for estimating the jammer subspace and computing the adapted weight vector. The set of M orthonormal vectors, $\{\mathbf{v}_m\}$, spanning the interference subspace is generated directly from N sample vectors, $\{\mathbf{x}_n\}$, using the Gram–Schmidt process. The statistical independence of the coefficients, $j_i(t_n)$, ensures that the interference subspace will be spanned with approximately M=L sample vectors. We include within the algorithm a test to determine whether or not jamming is present; if the power in the first data vector falls below a preset threshold corresponding to less than 3 dB above receiver noise, $\mathbf{w}_a$ is set to $\mathbf{w}_s$ and the procedure terminates. The algorithm has seven steps:

Step 1 Initialize index, set $m = 0$

Step 2 $m = m + 1$, $\mathbf{u}_m = \mathbf{x}_m$

Step 3 IF $m = 1$, $\mathbf{u}_m' = \mathbf{u}_m$

ELSE

$$\mathbf{u}_m' = \mathbf{u}_m - \sum_{k=1}^{m-1} (\mathbf{v}_k^H \mathbf{u}_m) \mathbf{v}_k$$

Step 4 Calculate $\mathbf{v}_m$ as

$$\mathbf{v}_m' \begin{cases} = \mathbf{u}_m' / |\mathbf{u}_m'| & |\mathbf{u}_m'|^2 > \Delta \\ = 0 & |\mathbf{u}_m'|^2 \leq \Delta \end{cases} \qquad \Delta = 2K\sigma^2$$

IF $\mathbf{v}_1 = 0$, DECIDE NO JAMMING, SET $\mathbf{w}_a = \mathbf{w}_s$, GO TO END

ELSE

Step 5 Calculate the statistic T as

$$T = \frac{\det(\mathbf{V}^H \mathbf{V})]^{\frac{1}{2k}}}{\prod_{i=1}^{k} |\mathbf{v}_i|^2}$$

where V comprises the $k \leq m$ non zero basis vectors as follows:

$$\mathbf{V} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k]$$

Step 6 If $T > \Delta_T$ GO TO Step 2

ELSE

a) Calculate w_a' as

$$w_a' = w_s - \sum_{m=1}^{M} (v_m^H w_s)\, v_m$$

b) Normalize as

$$w_a = w_a' \,/\, |\, w_a'|$$

END

The algorithm has two thresholds, an internal threshold, Δ, which rejects bad data vectors and an external threshold, Δ_T, which forces the continued collection of data vectors and provides a termination criterion for the algorithm.

The use of the statistic, T, and its comparison to an external threshold, Δ_T, as shown in step 5 of the algorithm, was proposed by Nickel [8]. By adding this procedure Nickel has converted the DVO method of Hung and Turner [3] into a data selection technique, albeit using a selection process which is less efficient than that of the DVSO method described in this report.

It is observed that T is a decreasing function of k where k is the number of non zero basis vectors selected in the algorithm at the m^{th} pass ($k \leq m$). As k increases, T decreases towards the threshold Δ_T; when k is large enough, say k=M, T falls below the threshold and the algorithm terminates. In general, the lower the value of Δ_T, the larger the value of M and the larger the number of data vectors that are considered. Thus, lowering Δ_T increases the SNJR up to a limiting value beyond which no improvement is observed. As well the computational load increases as Δ_T decreases.

4.1 COMPUTATIONAL COUNT

We first assume that N data vectors are used to compute M orthonormal vectors spanning the jammer subspace. An upper bound to the number of computations required by the DVO technique is then $(N^2 + 2.5M + .5)K + 0.5(N-1)$ complex multiplications, $(N^2 + 1.5M - 0.5)K - 0.5(N-1)$ complex additions, $(M+N)$ divisions and $(M+1)$ square roots. Step 3 requires a variable number of computations lying between $0.5M(M+1)$ and $0.5\,N(N-1)$ complex multiplications; therefore, the number of computations is a random variable for which the expectation can only be estimated by means of Monte Carlo simulations. A large external threshold, Δ_T, causes the process to terminate quickly with a moderately high jamming residue. Conversely, a small Δ_T causes the process to continue longer, to consider more data vectors and to improve the estimate of the jammer space; this results in lower jamming residues, a higher SNJR and a higher computational load.

The complex multiplications are the dominant computational operations. As a comparison, the SMI method requires $S(K+1)K/2 + K^3/2 + 2K^2$ complex multiplications where S is the number of samples used to form the covariance matrix. For example, let S=2K, M=5, N=7 and K=50. The upper bound for the GS orthogonalization is only 2855 complex multiplications as opposed to 195,000 for the SMI method. This decrease in computational load results in a lower SNJR than is achieved with the SMI technique. The trade–off in performance versus computational load will be examined in the following section.

4.2 COMPUTER SIMULATIONS

Computer simulations have been performed to analyze the effect of different parameters such as the number of jammers, the jammer strength, and the jammer direction on the performance of the DVO method.

We first analyze the effect of the number of jammers using test case T_1. Figure 1 shows the average SNJR as a function of Δ_T for L=1, 3 and 5. As the number of jammers, L, increases, the average SNJR decreases. Moreover, as the threshold, Δ_T, decreases, the SNJR increases. We also note that there is a very small variation in SNJR (less than 1 dB) over the entire range of variation of the threshold, Δ_T, in Fig. 1. We note that, in general, the optimum SNJR is a decreasing function of L; Figure 2 plots the difference between the optimum and average SNJRs to show the influence of an increasing number of jammers. For small values of L the SNJR is only few dBs (2–3 dB) below the SNJR optimum. As L increases, the SNJR deviates further from the SNJR optimum.

The external threshold, Δ_T, is directly related to the number of complex multiplications. Therefore, by varying this threshold, we can generate curves of SNJR as a function of the number of complex multiplications. Figure 3 shows the upper limit and the exact number of complex multiplications versus the threshold for L=3. As Δ_T increases, the difference between the curves decreases. Setting Δ_T at a high value limits the search to a small number of data vectors. Figure 4 shows the average SNJR versus the number of complex multiplications required for test case T_1 while figure 5 shows the improvement factor versus the same quantity. As the number of jammers, L, increases, the achieved SNJR decreases and the number of complex multiplications increases. Since the SNJR is a weak function of Δ_T, it is advantageous to choose a rather high Δ_T to achieve a good trade off between SNJR and the number of complex multiplications. From Figure 5 we see that the improvement factor increases slightly as L decreases.

Computer simulations have been performed to analyze the effect of the jammer strength using test case T_2. Figure 6 shows the average SNJR as a function of Δ_T for three equi–powered jammers with JNR's of 10, 20, 30 and 40 dB. As the JNR increases, the SNJR decreases. In contrast, however, the improvement factor (IF) increases with JNR; this is shown in figure 7 where IF is plotted versus the number of complex multiplications for different values of JNR. This follows because: (1) the jammer space can be more accurately estimated when JNR is high; and (2) the scope for improvement is much greater when JNR is high. The decrease in the absolute value of SNJR as JNR increases, reflects the fact that larger jammers lead to larger jamming residues. While the jammer space is more accurately estimated with larger values of JNR, this estimate is not sufficiently improved to offset the effect of larger residues. The decrease in SNJR as JNR increases is specific to the DVO technique; it is not observed for the other techniques to be described.

Finally, computer simulations have been performed for various jammer directions using test case T_3 comprising three sidelobe jamming and one mainbeam jamming scenario. Figure 8 shows the average SNJR versus complex multiplications for the sidelobe jamming scenarios D_1, D_2 and D_4. The DVO method provides essentially the same SNJR for the different jammer scenarios. Figure 9 gives the results for the mainbeam jamming scenario, D_3; here, although the SNJR at the output is less than 0 dB, we get an improvement over the

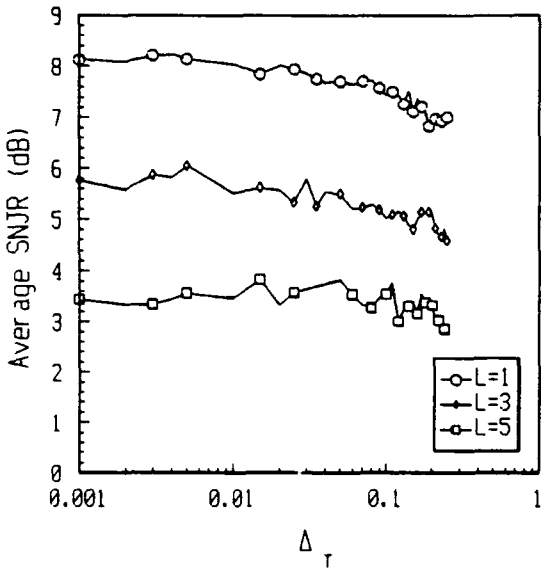

Fig. 1 Average SNJR versus Δ_T for the DVO method for K=10. Monte-Carlo simulations with 100 trials.

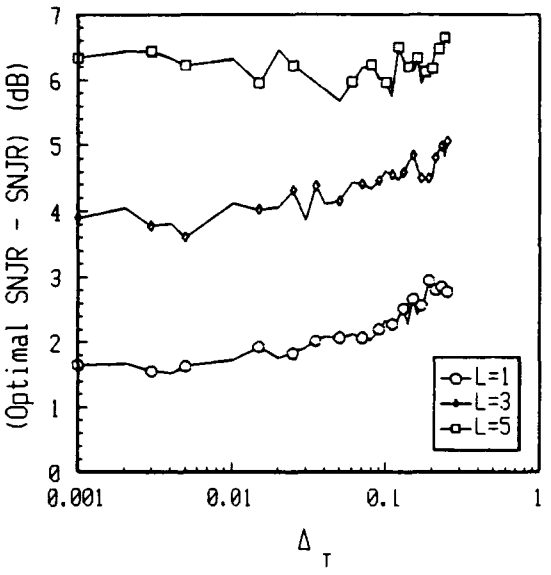

Fig. 2 (Optimal SNJR - SNJR) versus Δ_T for the DVO method for K=10. Monte-Carlo simulations with 100 trials.

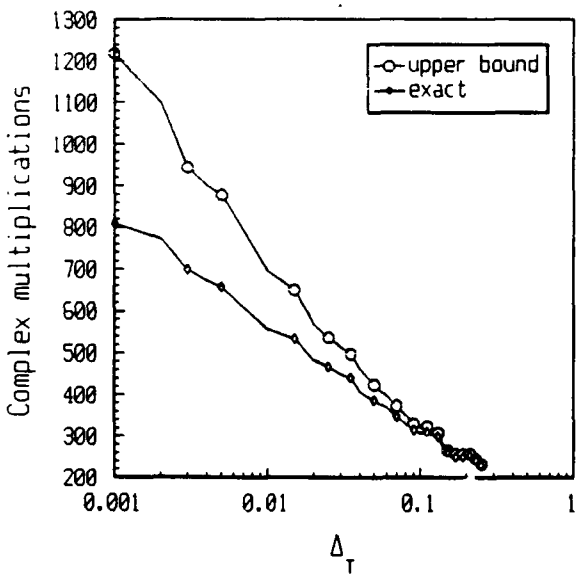

Fig. 3 Complex multiplications versus Δ_T for the the DVO method for K=10 and L=3. Monte-Carlo simulations with 100 trials.

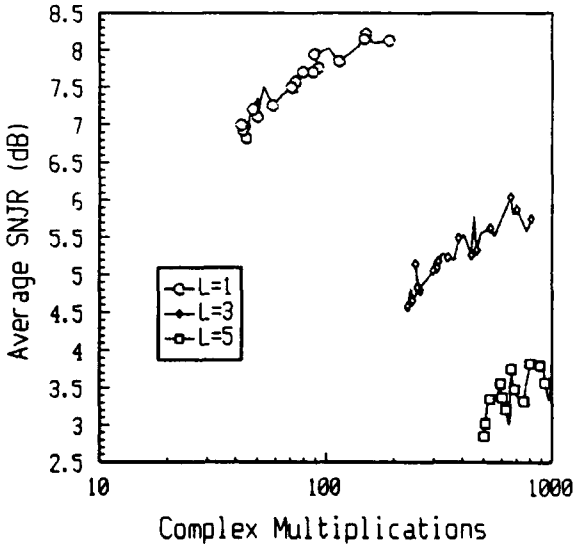

Fig. 4 Average SNJR versus complex multiplications for the DVO method with K=10. Monte-Carlo simulations with 100 trials.

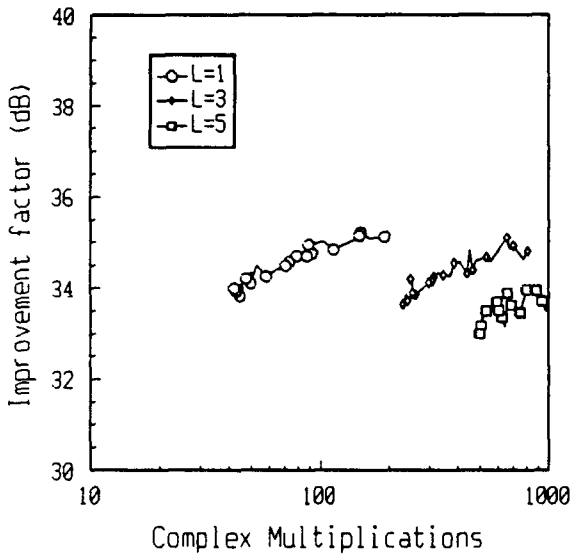

Fig. 5 Improvement factor versus complex multiplications for the DVO method with K=10. Monte-Carlo simulations with 100 trials.

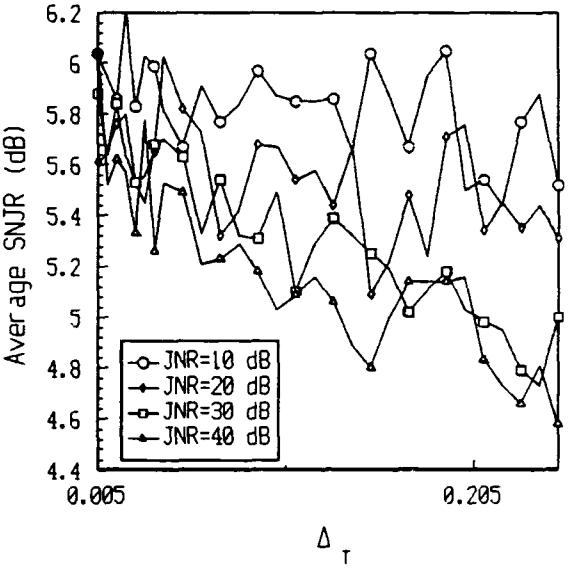

Fig.6 Average SNJR versus Δ_T for the DVO method with K=10. Monte-Carlo simulations with 100 trials.

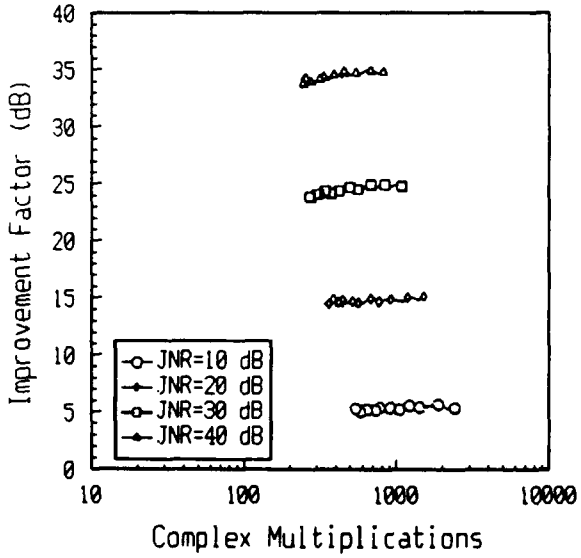

Fig. 7 Improvement Factor versus complex multiplications for the DVO method with K=10. Monte-Carlo simulations with 100 trials.

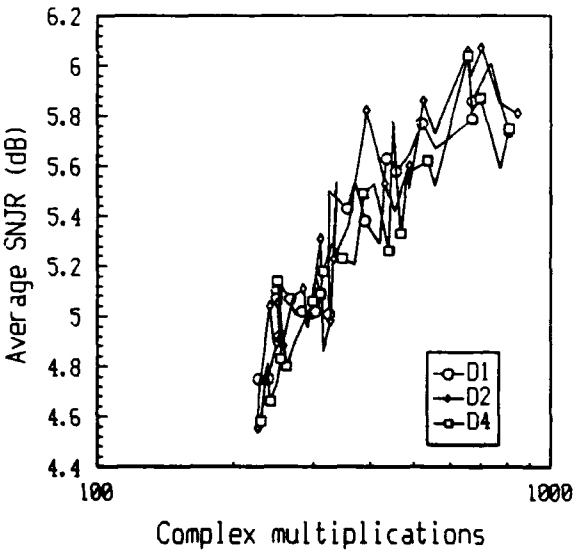

Fig 8 Average SNJR versus complex multiplications for the DVO method with K=10 for different jammer scenarios. Monte-Carlo simulations with 100 trials.

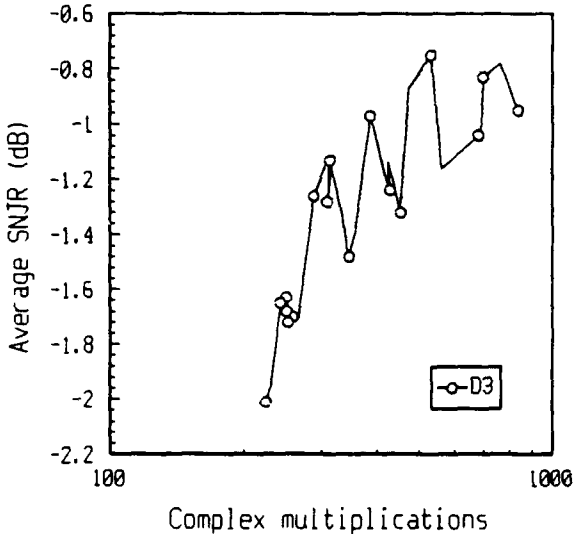

Fig 9 Average SNJR versus complex multiplications for the DVO method with K=10 for a mainbeam jamming scenario Monte-Carlo simulations with 100 trials.

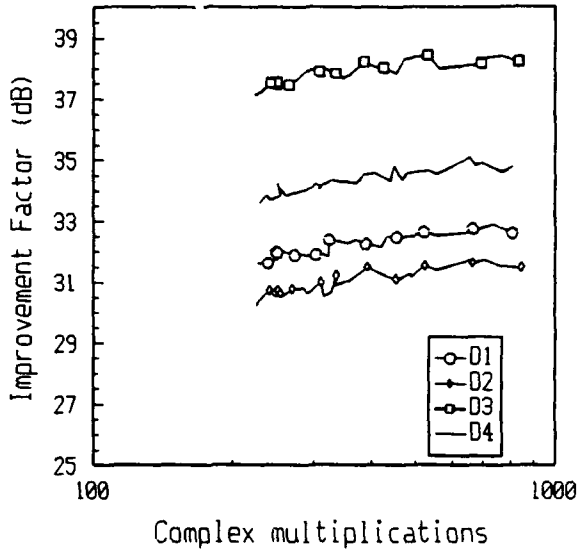

Fig 10 Improvement factor versus complex multiplications for the DVO method with K=10 for different jammer scenarios. Monte-Carlo simulations with 100 trials.

unadapted situation. Figure 10 plots the improvement factor versus complex multiplications for the four jammer scenarios. We see that the improvement factor for the sidelobe jamming situations has a spread of about 4 dB. Although the average SNJR for the mainbeam jamming situation is less then 0 dB, a improvement factor of about 38 dB is achieved; this is better than that achieved for the sidelobe jamming scenario.

The DVO method is a very fast method to obtain the adapted weight vector. The price paid for the small computational load is a lower SNJR than is achievable with the SMI technique. A modification to the DVO technique is proposed; by applying a data selection process prior to GS orthogonalization, we trade off some computational efficiency to obtain an improved SNJR. This new method is called Data Vector Selection and Orthogonalization.

5.0 DATA VECTOR SELECTION AND ORTHOGONALIZATION

The basis of the DVSO technique is the selection of the best M vectors from a larger group of N vectors which are sampled when the signal is absent and stored before the process begins. The selected M vectors are best in the sense that they yield the highest SNJR as compared with any other set of M vectors selected from the larger set of N. The selection process, the GS orthogonalization and the computation of the adapted weight, $\mathbf{w}_a$, are combined in a single procedure.

In the algorithm for determining $\mathbf{w}_a$, we use the following notation: $\{\mathbf{x}_n\}$ comprises the set of N sample vectors, $\{\mathbf{u}_m\}$ denotes the M best sample vectors chosen from the $\{\mathbf{x}_n\}$, and $\{\mathbf{v}_m\}$ denotes the orthonormal basis formed from the $\{\mathbf{u}_m\}$ by the Gram–Schmidt process. The algorithm consists of nine steps:

Step 1 N snapshots $\{\mathbf{x}_n\}$, n=1...N, are taken when the signal is absent. Each snapshot comprises a sum of the jammers and receiver noise. Calculate $I_n^1 = |\mathbf{x}_n|^2$ for n = 1...N. Set $\mathbf{u}_1 = \mathbf{x}_n$ for n which gives a maximum I_n^1.

Step 2 Calculate $\mathbf{v}_1$ as

$$\mathbf{v}_1 = \begin{cases} \mathbf{u}_1/|\mathbf{u}_1| & |\mathbf{u}_1|^2 > \Delta \\ \mathbf{0} & |\mathbf{u}_1|^2 \leq \Delta \end{cases} \qquad \Delta = 2K\sigma^2$$

IF $\mathbf{v}_1 = 0$, DECIDE NO JAMMING, SET $\mathbf{w}_a = \mathbf{w}_s$, GO TO END

The objective of this thresholding operation is to reject sample vectors which are within 3 dB of the receiver noise. If $\mathbf{v}_1 = 0$, this means that the power in every data vector is within 3 dB of the receiver noise power; therefore it is decided that no significant jamming is present and the algorithm terminates.

Step 3 Calculate $\mathbf{w}_a^1$ as

$$\mathbf{w}_a^1 = \mathbf{w}_s - (\mathbf{v}_1^H \mathbf{w}_s)\mathbf{v}_1$$

where $\mathbf{w}_s$ is the weight vector specifying the steering direction for the array.

Step 4 m=2

Step 5 Determination of $\mathbf{u}_m$.

Calculate $I_n^m = |(\mathbf{w}_a^{m-1})^H \mathbf{x}_n|^2$ for n=1...N. Set $\mathbf{u}_m = \mathbf{x}_n$ for n which gives a maximum I_n^m.

Step 6 Calculate $\mathbf{u}_m'$ as

$$\mathbf{u}_m' = \mathbf{u}_m - \sum_{k=1}^{m-1} (\mathbf{v}_k^H \mathbf{u}_m) \mathbf{v}_k$$

Step 7 Calculate $|\mathbf{u}_m'|^2$

Step 8 Calculate $\mathbf{v}_m$ as

$$\mathbf{v}_m = \begin{cases} \mathbf{u}_m'/|\mathbf{u}_m'| & |\mathbf{u}_m'|^2 > \Delta \\ 0 & |\mathbf{u}_m'|^2 \leq \Delta \end{cases}$$

Step 9 If $\mathbf{v}_m \neq 0$

a) Calculate $\mathbf{w}_a^m = \mathbf{w}_a^{m-1} - (\mathbf{v}_m^H \mathbf{w}_a^{m-1}) \mathbf{v}_m$

b) m+1

c) Return to step 5

else

d)Calculate $\mathbf{w}_a$ as

$$\mathbf{w}_a = \mathbf{w}_a^m / |\mathbf{w}_a^m|$$

e) End

The purpose of the threshold, Δ, is twofold: to ensure that vectors associated with receiver noise are discarded and to provide a means of terminating the algorithm.

The SNJR at the output of the array depends on the choice of N. As N becomes large, the SNJR approaches an asymptote; this asymptote is lower than the optimum SNJR which is approached asymptotically by the SMI method as the number of sample vectors becomes large.

5.1 COMPUTATIONAL COUNT

For an iterative process terminating with M basis vectors, the DVSO algorithm requires

$$C(M, K, N) = [K(1.5+N/2+4M+MN+M^2)+MN/2]$$

complex multiplications. In addition, we require $[K(-0.5+N/2+3M+MN+M^2)]$ complex additions, M+2 divisions and M+2 square roots. If $M << K$, significant computational savings over SMI method can be made. In the example given previously for the DVO technique (M=5, K=10), if N=2M, the DVSO method requires 5,100 complex multiplications which is less than twice the number of complex multiplications required by the DVO method and far less than the 195,000 complex multiplications required by the SMI method. Compared with the basic DVO algorithm, the DVSO version is more computationally demanding; it must be realized, however, that the SNJR achieved is better than that of the basic DVO algorithm.

5.2 COMPUTER SIMULATIONS

We first analyze the effect of the number of jammers on the performance of the DVSO method using test case T_1. Figure 11 plots the average SNJR versus N. As N increases, the average SNJR increases to a limiting value for a fixed L. The SNJR is fairly sensitive to N which must be chosen large enough to provide an acceptable SNJR. We also note that the performance is quite sensitive to the number of jammers. As the number of jammers, L, increases, the SNJR decreases. For one jammer (L=1), an SNJR value very close to optimum is achieved with a small calculation load as shown in figure 12. Figure 13 shows the improvement factor as a function of the number of complex multiplications. As the number of complex multiplications becomes large, all of the curves for the improvement factor approach an asymptote as shown in Figure 13.

Figure 14 illustrates the behavior of the algorithm for various JNR with three equi–powered jammers using test case T_2. In contrast to the DVO method, as JNR increases, it is possible to obtain an increased SNJR but at the expense of a higher computational load as shown in Figure 14. We note that for very high values of JNR, i.e. JNR=30 dB and JNR=40 dB, the SNJR curves exactly coincide. Figure 15 gives the corresponding improvement factors as a function of N with JNR as a parameter. As expected, the improvement factor increases with JNR.

Figure 16 gives the curves for SNJR versus complex multiplications for different jammer directions for the sidelobe jamming scenario. We see that the performance of the algorithm doesn't vary with the jammer direction provided the jammers are in the array side lobe region.

Results for the mainbeam jamming scenario, D_3, are shown in figure 17. The parameter N must be chosen high enough to allow the method to null the jammers. In the scenario shown, the SNJR is positive for $N \geq 12$. The SNJR achieved is small for the mainbeam jamming scenario (approximately .2 dB to .3 dB); it must be remembered, however, that this SNJR is calculated assuming 0 dB SNR at the element output. Therefore, any increase of the SNR at the element level, adds, dB–wise, to the SNJR at the array output. Such an increase can be achieved by coherent or non–coherent integration, and/or by designing a system with a larger power aperture product so that a SNR > 0 dB is achieved at the element level. The important point is that the SNJR is vastly better than if no adaptive nulling was used (SNJR=–39.05 dB in this case). For a 10–element array, the maximum achievable SNJR under this assumption is 10 dB. Adaptive nulling gives approximately 50 dB of jammer suppression in the mainbeam at the expense of a 10 dB signal loss for an overall improvement factor of 40 dB – a very useful performance in most circumstances.

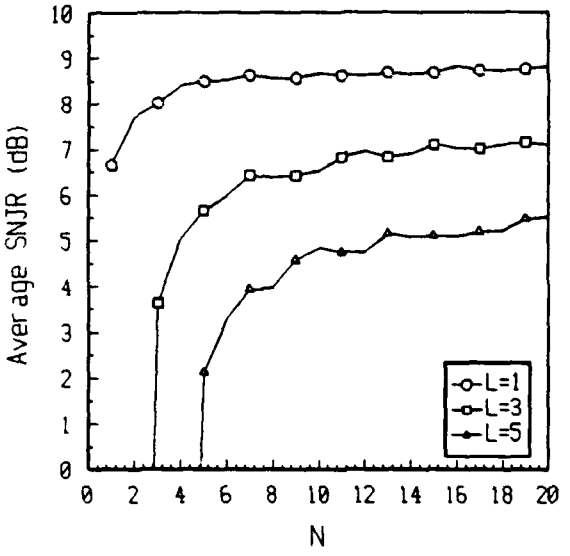

Fig. 11 Average SNJR versus N for the DVSO method for L=1, 3 and 5. Monte-Carlo simulations with 100 trials.

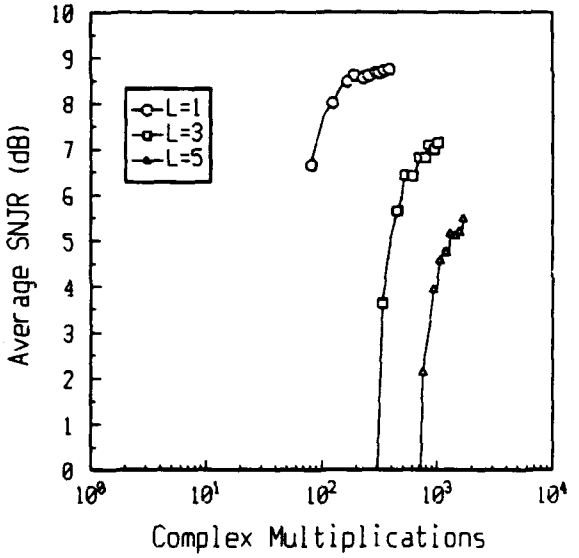

Fig 12 Average SNJR versus complex multiplications for DVSO method for L=1, 3 and 5. Monte-Carlo simulations with 100 trials.

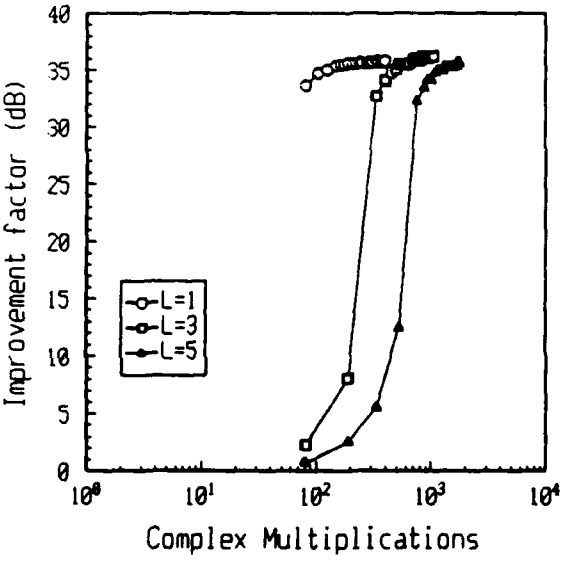

Fig 13 Improvement factor versus complex multiplications for DVSO method for L=1, 3 and 5. Monte-Carlo simulations with 100 trials.

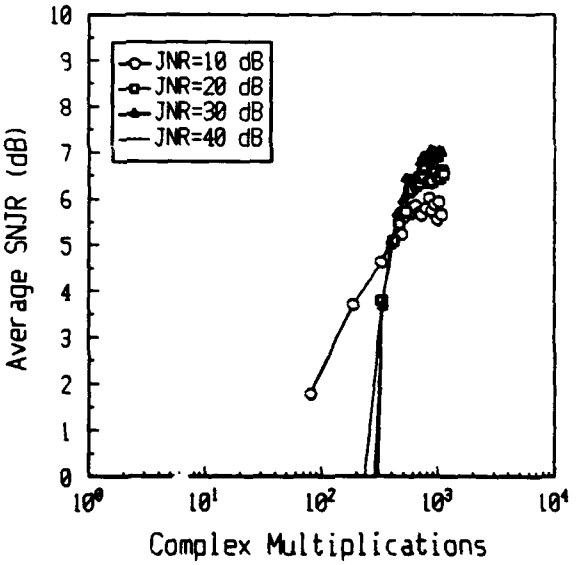

Fig. 14 Average SNJR versus complex multiplications for for the DVSO method for K=10 and L=3. Monte-Carlo simulations with 100 trials.

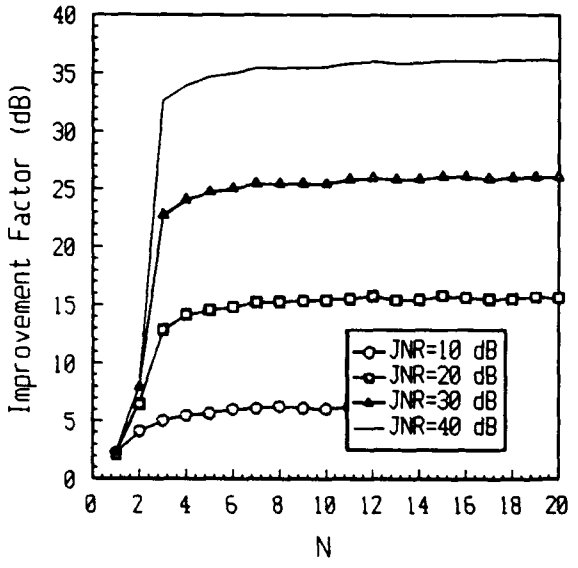

Fig. 15 Improvement Factor versus N for the DVSO method for K=10 and L=3. Monte-Carlo simulations with 100 trials.

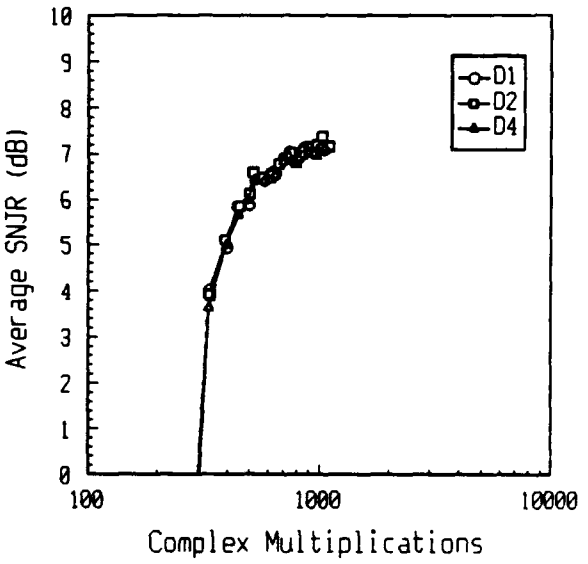

Fig. 16 Average SNJR versus complex multiplications for the DVSO method with K=10, L=3. Monte-Carlo simulations with 100 trials.

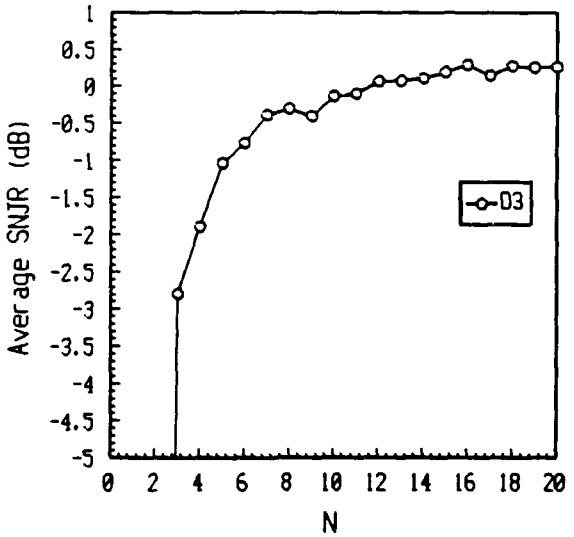

Fig. 17 Average SNJR versus N for the DVSO method with K=10, L=3 in a mainbeam jamming situation. Monte-Carlo simulations with 100 trials.

6.0 COMPARISON OF DVO AND DVSO

Both the DVSO and DVO methods combine GS orthogonalization and computation of the adapted weight vector with a data selection process. In the DVO method, the data selection process consists of eliminating those data vectors associated with receiver noise while continuing to take new sample vectors until a second, external threshold criterion is satisfied. The selection is carried out after the GS orthogonalization. In the DVSO technique, the data selection process consists of choosing the data vector that correlates the most strongly with the current adapted weight vector; this occurs prior to GS orthogonalization. The data selection process of the DVO method is not as efficient as that of the DVSO method. In the DVO method, we eliminate data vectors that are not good enough while in DVSO we select the best data vectors. Therefore the DVSO method gives better results than DVO method in terms of SNJR; the data selection criterion, however, requires more computations.

Figures 18 to 20 plot average SNJR versus complex multiplications for an array of 10 elements with L jammers for L=1, 3, 5 respectively. The DVSO method provides a higher SNJR than DVO but at the expense of a higher computational load. Note also that as L increases, the difference between DVO and DVSO increases: for L=1, the difference is $\approx$ .8 dB while for L=5 the difference is $\approx$ 1.7 dB.

The DVSO and DVO methods require far fewer computations than does the SMI method but at the expense of a smaller SNJR. The SMI method benefits from averaging the data in the formation of the covariance matrix which reduces the effect of receiver noise and jammer variability. In the following we present two new methods for incorporating averaging techniques into DVSO and improving the SNJR: the DVSO–WAVER and the DVSO–COVAR techniques.

7.0 INCORPORATING AVERAGING TECHNIQUES

7.1 DVSO WITH WEIGHT–VECTOR AVERAGING

In this technique we average repeated computations of the adapted weight vector each obtained by using the DVSO method. We introduce a new parameter, Q, the number of adapted weight vectors averaged. The new adapted weight vector, $\mathbf{w}_a$, is given by

$$\mathbf{w}_a = \frac{1}{Q} \sum_{i=1}^{Q} \mathbf{w}_{ai}$$

where the $\mathbf{w}_{ai}$ are the adapted weight vectors obtained with the DVSO technique. Since this method involves weight–vector averaging, it is called the DVSO–WAVER technique.

The computational load of the DVSO–WAVER technique is the sum of the computational load of the DVSO technique for each adapted weight vector. We define M_i as the number of basis vectors used in computing $\mathbf{w}_{ai}$. M_i is now a random variable with a unknown probability function. The total number of complex multiplications is obtained by summing those for the computation of each $\mathbf{w}_{ai}$ as

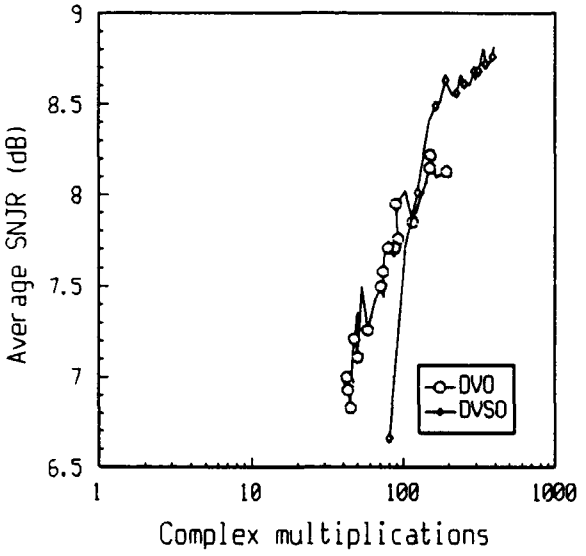

Fig 18 Average SNJR versus complex multiplications for the DVO and DVSO methods with K=10 and L=1. Monte-Carlo simulations with 100 trials.

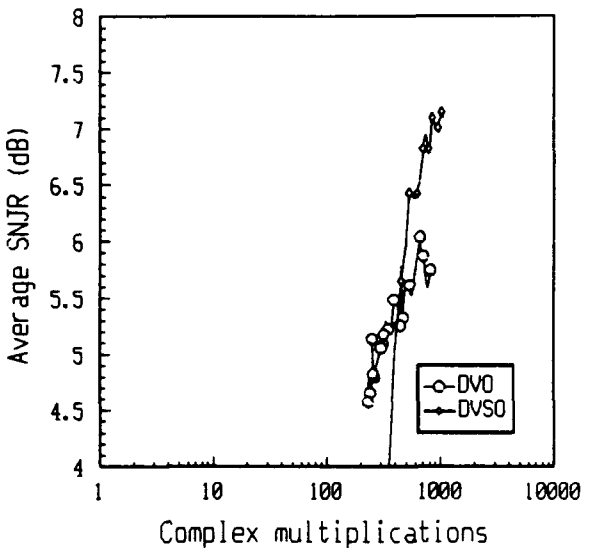

Fig 19 Average SNJR versus complex multiplications for the DVO and DVSO methods with K=10 and L=3. Monte-Carlo simulations with 100 trials.

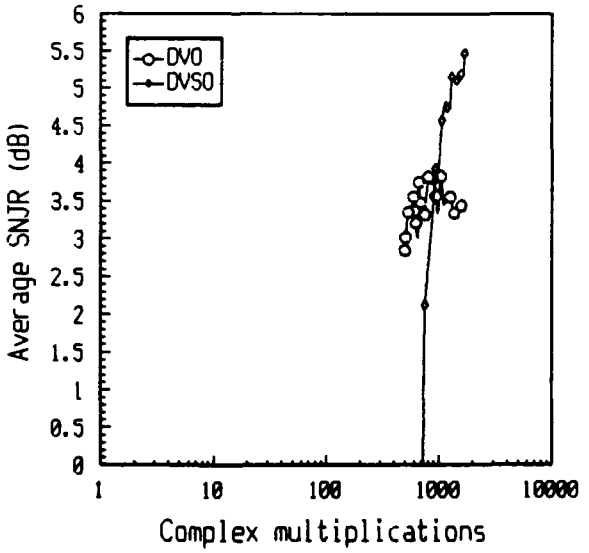

Fig 20 Average SNJR versus complex multiplications for the DVO and DVSO methods with K=10 and L=5. Monte-Carlo simulations with 100 trials.

$$\sum_{i=1}^{Q} C(M_i, K, N)$$

where $C(M_i, K, N)$, as defined for the DVSO technique in the previous section, is the number of complex multiplications required to compute the i^{th} adapted weight vector for an array of K elements with M_i data vectors from a set of N data vectors. Since M_i is a random variable, the average computational load is estimated by means of Monte–Carlo simulations, using repeated trials with different samples of the noise and jamming.

7.1.1 COMPUTER SIMULATIONS

Computer simulations have been performed to demonstrate the effect of the number of jammers, L, on the performance of the DVSO–WAVER method using test case T_1. Figure 21 plots the average SNJR versus Q for an array of 10 elements with N=9 for L=1, 3 and 5. For each scenario, the curve for SNJR approaches an asymptote which is below the optimal SNJR. In figure 22, we plot the difference between the optimal and actual SNJR values versus Q for the different values of L; the difference decreases as the number of jammers decreases. The best performance is achieved when the number of jammers is small.

Figure 23 gives the curves of SNJR versus complex multiplications for the three jammer scenarios. As L increases, the SNJR decreases and the number of complex multiplications increases. Figure 24 shows the improvement factor versus complex multiplications. Provided that Q and hence the number of complex multiplications are sufficiently large, the improvement factor increases as the number of jammers, L, increases.

We next analyze the effect of varying the strength of jammers using test case T_2. Figure 25 shows the improvement factor versus complex multiplications while Figure 26 gives the SNJR versus complex multiplications. As expected, both the improvement factor and the SNJR increase with increasing JNR.

Finally, computer simulations have been performed for various jammer directions using test case T_3. Figure 27 plots the average SNJR versus Q for the sidelobe jamming scenarios D_1, D_2 and D_4 obtained with N=9. We see a slight variation in behavior for different jammer directions. Since the optimum SNJR varies according to the jammer direction, we have plotted in Figure 28 the difference between the optimum SNJR and the average SNJR versus Q for the four different scenarios. The curves for D_1, D_2 and D_4 lie nearly on the top of each other. Figure 29 gives similar curves for SNJR versus the number of complex multiplications for the sidelobe jamming scenarios, D_1, D_2, and D_4. As in Fig. 27, results for the sidelobe jamming scenarios are very similar. In contrast, Figure 30, which gives the improvement factors versus complex multiplications, shows significant differences for both the sidelobe and mainlobe jamming scenarios. Again the largest improvement factor of about 41 dB is observed for the mainlobe jamming scenario, D_3. Improvement factors of 32–38 dB were observed for the sidelobe jamming scenarios.

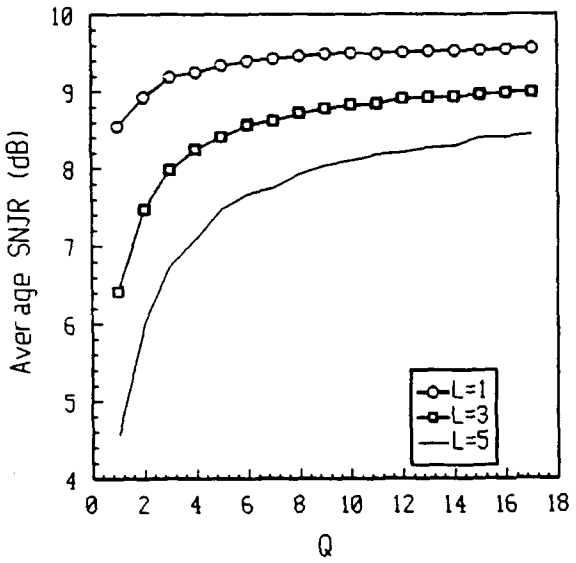

Fig.21 Average SNJR versus Q for the DVSO-WAVER method with K=10 and N=9. Monte-Carlo simulations with 100 trials.

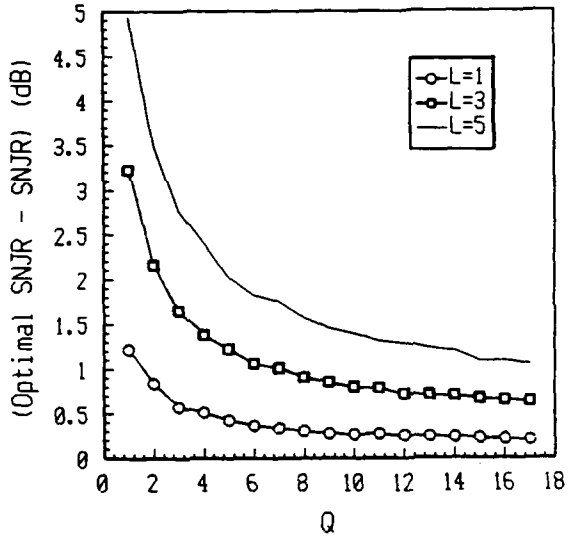

Fig.22 (Optimal SNJR - SNJR) versus Q for the DVSO-WAVER method for K=10 and N=9. Monte-Carlo simulations with 100 trials.

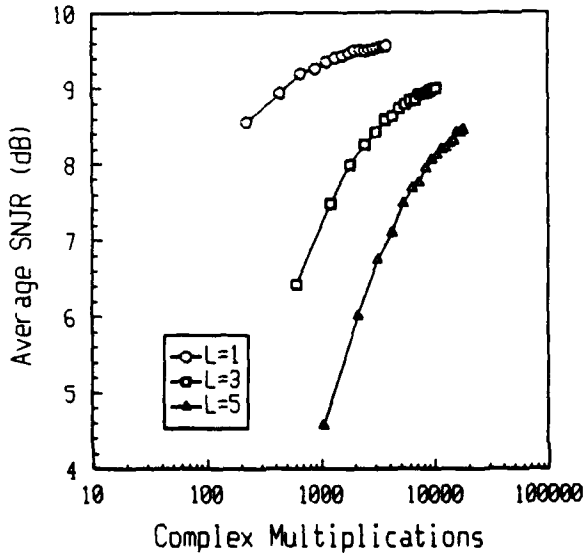

Fig.23 Average SNJR versus complex multiplications for the DVSO-WAVER for K=10 and L=1, 3, 5. Monte-Carlo simulations with 100 trials.

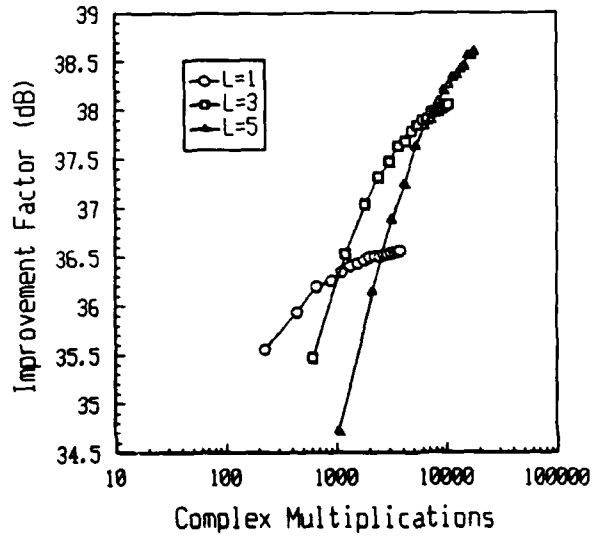

Fig 24 Improvement factor versus complex multiplications for the DVSO-WAVER for K=10 and L=1, 3, 5. Monte-Carlo simulations with 100 trials.

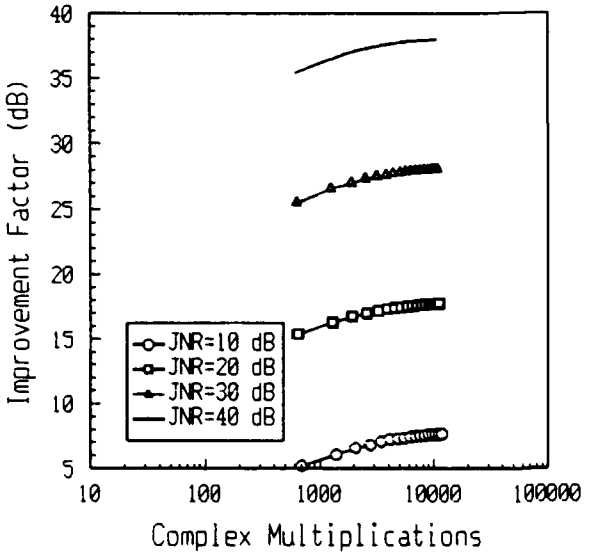

Fig 25 Improvement Factor versus complex multiplications for the DVSO-WAVER method for K=10, L=3 and N=9. Monte-Carlo simulations with 100 trials.

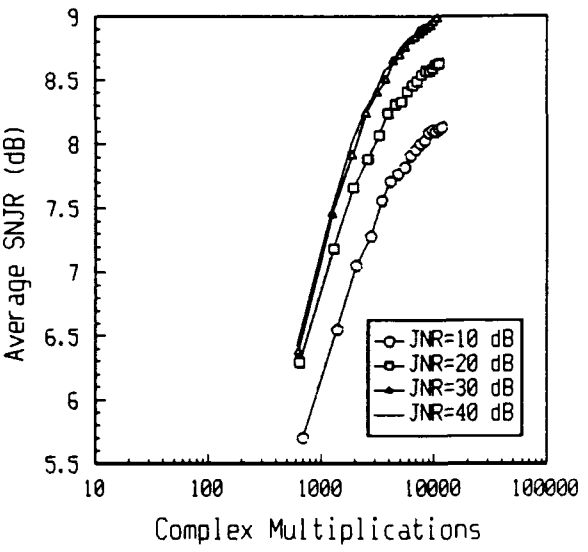

Fig. 26 Average SNJR versus complex multiplications for the DVSO-WAVER method for K=10, L=3 and N=9. Monte-Carlo simulations with 100 trials.

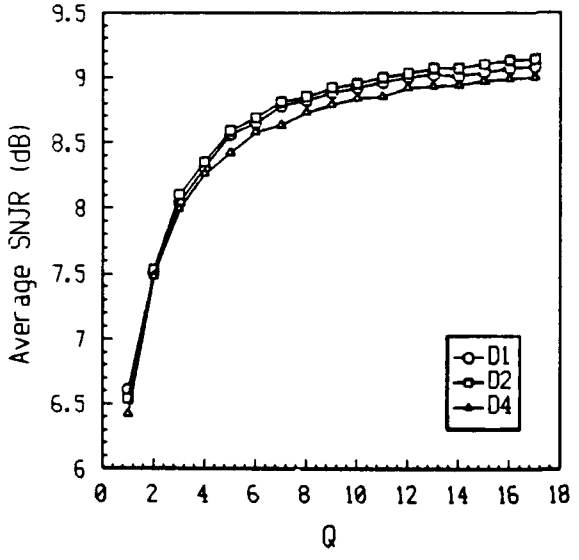

Fig 27 Average SNJR versus Q for the DVSO-WAVER method for K=10, L=3 and N=9. Monte-Carlo simulations with 100 trials.

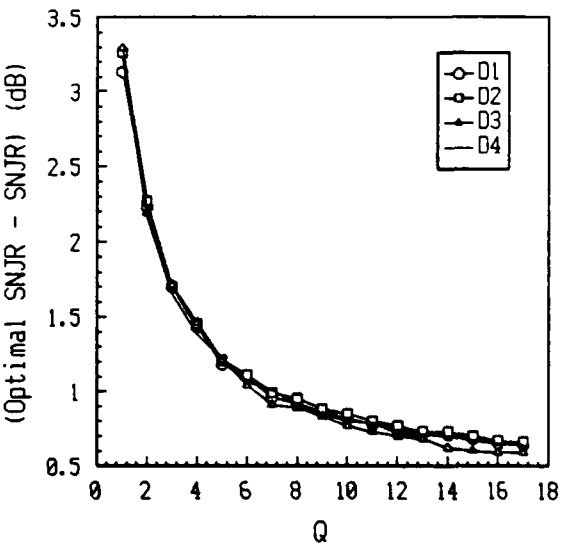

Fig 28 (Optimal SNJR - SNJR) versus Q for the DVSO-WAVER for K=10, L=3 and N=9. Monte-Carlo simulations with 100 trials.

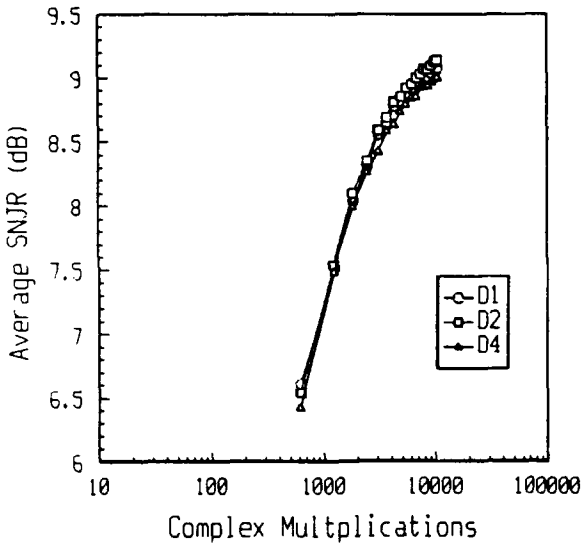

Fig 29 Average SNJR versus complex multiplications for the DVSO-WAVER method for K=10, L=3 and N=9. Monte-Carlo simulations with 100 trials.

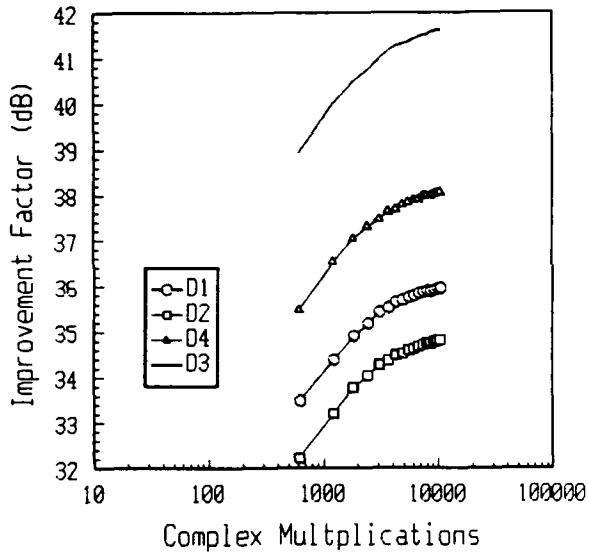

Fig 30 Improvement Factor versus complex multiplications for the DVSO-WAVER method for K=10, L=3 and N=9. Monte-Carlo simulations with 100 trials.

7.2 DVSO FOR ESTIMATING THE COVARIANCE MATRIX

This technique employs the DVSO method in calculating the covariance matrix and processing its columns to obtain an adapted weight vector. The major feature of this algorithm is its selection process: the best data vectors $\{\mathbf{u}_m\}$ are chosen to form the covariance matrix. We are thus able to obtain a good estimate of the covariance matrix using fewer but higher quality data vectors than in the conventional approach. The DVSO method is applied in order to find the M best data vectors out of a set of N vectors $\{\mathbf{x}_n\}$. This process is repeated Q times to give

$$M_T = \sum_{i=1}^{Q} M_i$$

data vectors which are used to calculate the covariance matrix R:

$$R = \frac{1}{M_T} \sum_{i=1}^{M_T} \mathbf{x}_i \, \mathbf{x}_i^H$$

The final step applies the DVSO method to the columns of the covariance matrix to find an adapted weight vector. This method is called DVSO–COVAR.

The number of computations required by the DVSO–COVAR technique is a random variable; the mean number of calculations is estimated by Monte–Carlo simulations. For a single trial evaluation of the adapted weight, the number of multiplications is

$$\sum_{i=1}^{Q} C(M_i, N, K) - KQ + M_T K (K + 1)/2 + C(H, K, K)$$

where H is the number of columns used as data vectors to calculate the adapted weight vector. The first three terms give the computations required for the data selection while the fourth term is associated with the calculation of the adapted weight vector. Note that since the covariance matrix has K columns, the parameter N is set to K in the final term. Increased efficiency may be achieved by using less than K columns. We have obtained some results showing that the algorithm achieves the same SNJR when we use only L columns where L is the number of jammers. This implies that L columns of the covariance matrix are sufficient to describe the jammer subspace. The value of L is generally unknown; however, we could choose N based on a rough estimate of the number of jammers and achieve some additional reduction of the computational load. This saving will be significant when $N << K$ as is often the case. This factor was not considered in the results presented in this report where the value of N was set to K.

The DVSO–COVAR method requires more computations than DVSO–WAVER for a given Q but gives a higher SNJR as will be shown.

7.2.1 COMPUTER SIMULATIONS

We first analyze the effects of having one, three and five jammers active using test case T_1. Figure 31 shows the variation of SNJR versus Q for L=1, 3 and 5 jammers with N=9. In all cases, the SNJR approaches an asymptote as Q increases; however, the limiting asymptote decreases (i.e. performance drops off) as the number of jammers increases. Figure 32 shows the difference (in dB) between the achieved SNJR and the optimal SNJR as a function of Q. In all cases, the difference appears to be approaching zero asymptotically but at a slower rate as the number of jammers increases.

We now look at the computational load required to achieve a specific SNJR. Figure 33 shows curves of SNJR versus the number of complex multiplications for test case T_1 while Figure 34 shows the improvement factor. As L increases, a higher computational load is required to achieve a given SNJR level and the limiting SNJR decreases.

Using test case T_2, we now examine the effect of the jammer–to–noise ratios, JNR, on performance for JNR values of 10, 20, 30 and 40 dB with N=9. Figure 35 shows the variation of SNJR with complex multiplications while Figure 36 shows the improvement factor for JNR's of 20, 30, 40 dB. The SNJR curves lie virtually on top of each other while Figure 36 indicates larger improvement factors are obtained as JNR increases. We note however that the technique did not work for JNR =10 dB i.e. an SNJR of about 1 dB was obtained. This was remedied by lowering the threshold of the DVSO technique when applied to the columns of the covariance matrix while keeping the same value for threshold for the selection of the data vectors. We designate the threshold for column selection as Δ_c and choose $\Delta_c = 0.2\ K\sigma^2$, i.e. $\Delta_c = \Delta/10$. Figure 37 shows the curve for the average SNJR versus complex multiplications for JNR=10 dB and compares this curve to the results previously obtained in Figure 35. We see that with this variation the algorithm works also for weak jammers.

Test case T_3 is used to examine performance as a function of jammer directions. We have three sidelobe jamming and one main lobe jamming scenarios. Figure 38 plots SNJR versus Q for the three sidelobe jamming scenarios while Figure 39 gives the curves for the difference between the optimal and the average SNJR (dB) versus Q for all four jamming scenarios with N=9. Figure 40 shows the number of computations versus SNJR for the sidelobe jamming scenarios.

7.3 DVSO–WAVER COMPARED WITH DVSO–COVAR

Figures 41 to 43 show the SNJR for DVSO–WAVER and DVSO–COVAR methods as a function of Q. We see that DVSO–COVAR always performs better for a fixed value of Q. Moreover, as L increases the difference in SNJR between the two algorithms increases – from about .1 dB for L=1 to 1 dB for L=5. Note that, for a given Q, the computational load of DVSO–COVAR is greater than that of DVSO–WAVER.

Finally, figure 44 gives simulation results for a mainbeam jamming situation for the DVSO–WAVER and DVSO–COVAR methods. The results of figure 44 are very similar to that observed for the sidelobe jamming situation.

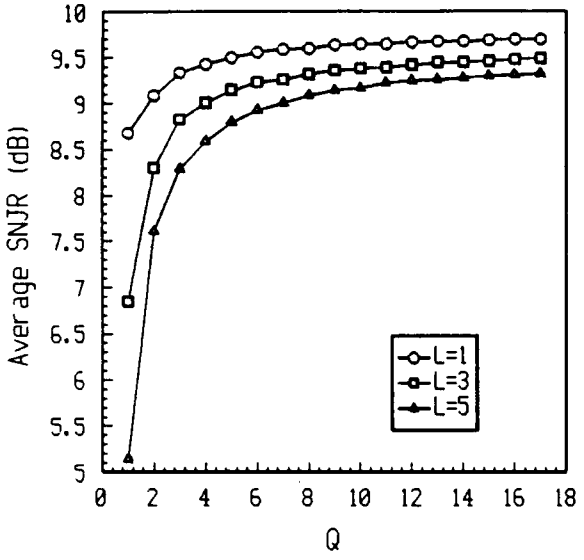

Fig. 31 Average SNJR versus Q for the DVSO-COVAR method with K=10 and N=9. Monte-Carlo simulations with 100 samples.

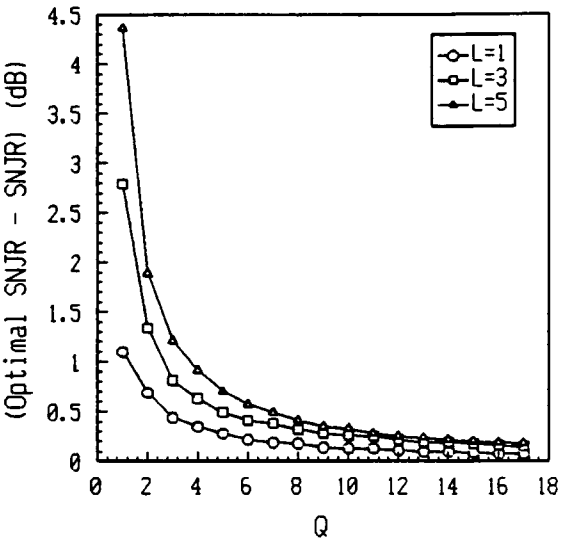

Fig 32 (Optimal SNJR - SNJR) versus Q for the DVSO-COVAR method with K=10 and N=9. Monte-Carlo simulations with 100 trials.

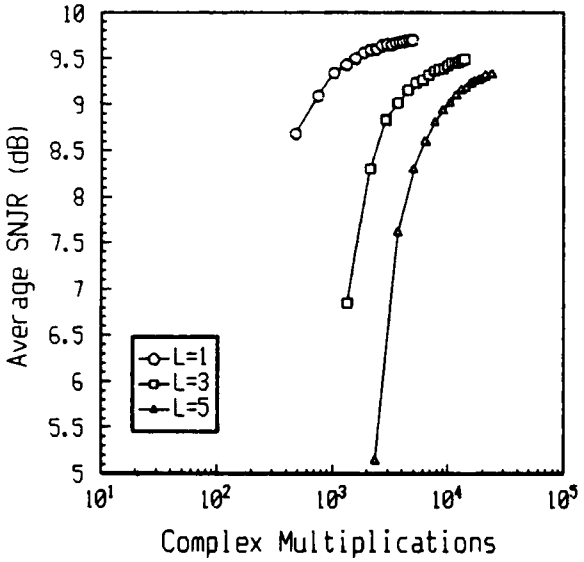

Fig 33 Average SNJR versus complex multiplications for for the DVSO-COVAR method for K=10 and N=9. Monte-Carlo simulations with 100 trials.

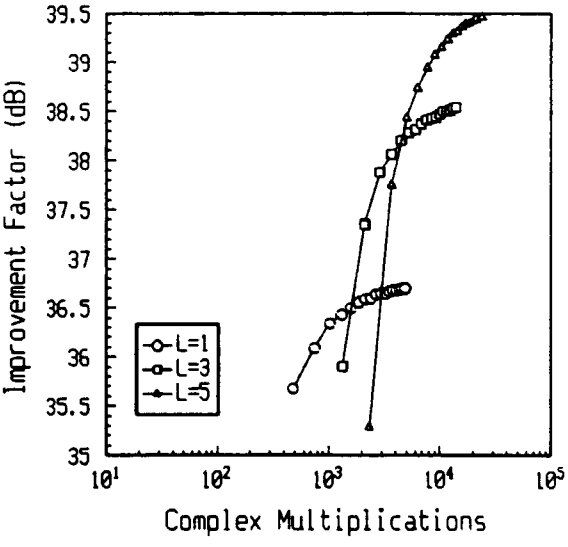

Fig 34 Improvement Factor versus complex multiplications for the DVSO-COVAR method for K=10 and N=9. Monte-Carlo simulations with 100 trials.

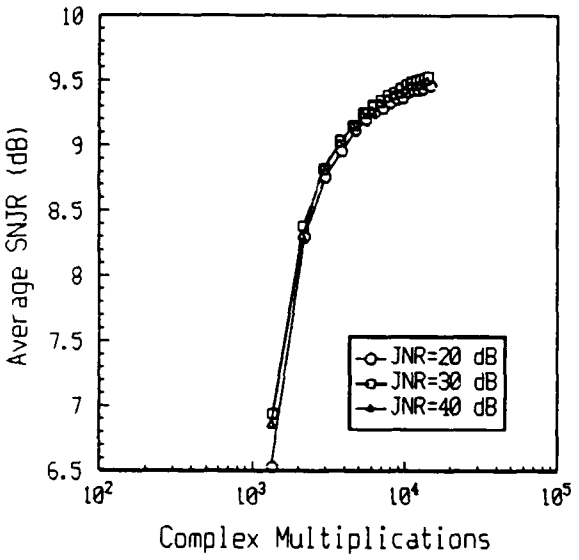

Fig. 35 Average SNJR versus complex multiplications for the DVSD-COVAR method for K=10, L=3 and N=9. Monte-Carlo simulations with 100 trials

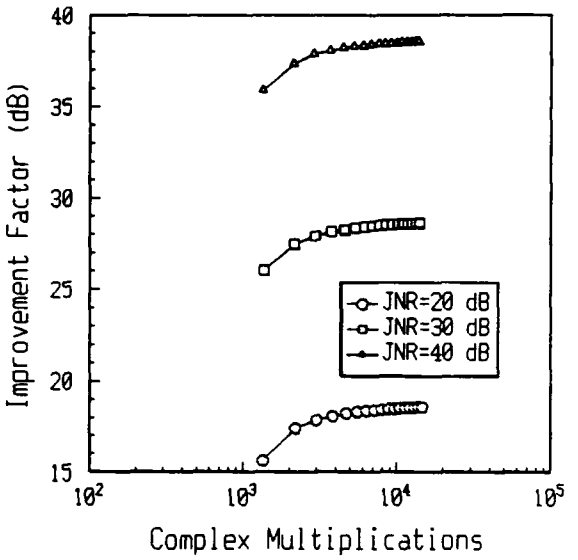

Fig 36 Improvement factor versus complex multiplications for the DVSD-COVAR method for K=10, L=3 and N=9. Monte-Carlo simulations with 100 trials.

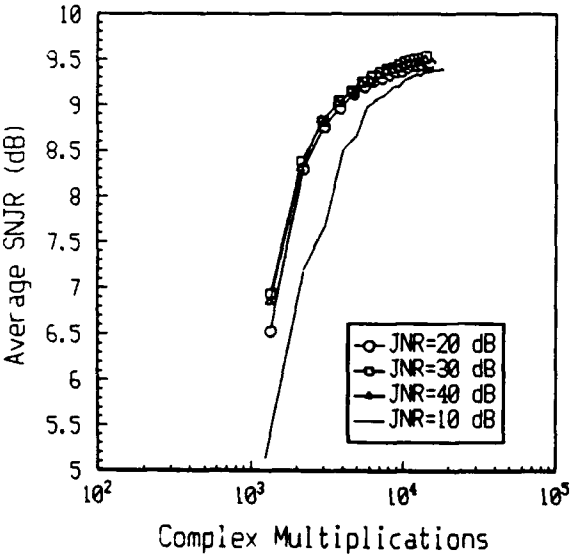

Fig. 37 Average SNJR versus complex multiplications for the DVSD-COVAR method for K=10, L=3 and N=9. Monte-Carlo simulations with 100 trials

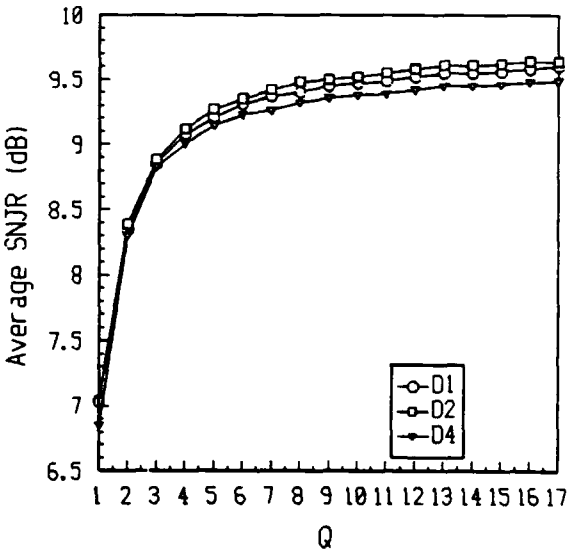

Fig 38 Average SNJR versus Q for the DVSD-COVAR method for K=10, L=3, N=9. Monte-Carlo simulations with 100 trials.

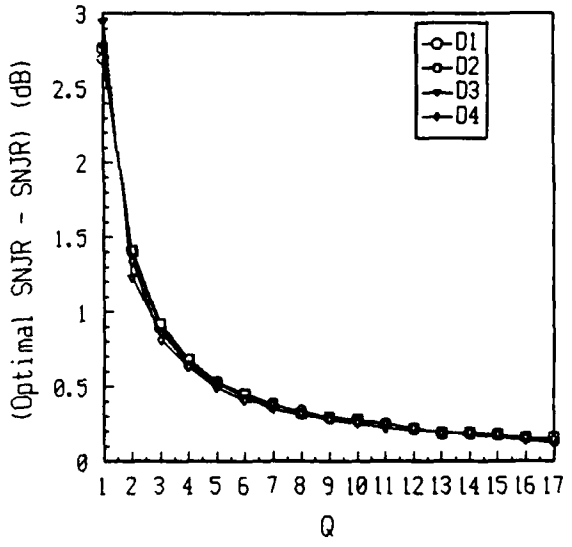

Fig 39 (Optimal SNJR - SNJR) versus Q for the DVSO-COVAR method for K=10, L=3 and N=9. Monte-Carlo simulations with 100 trials.

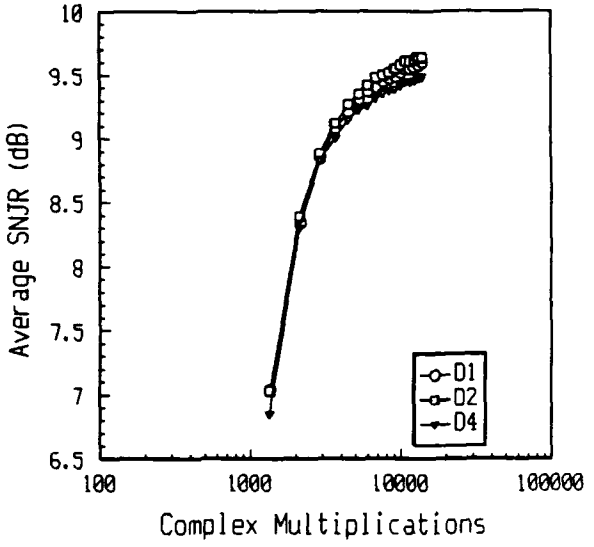

Fig. 40 Average SNJR versus complex multiplications for the DVSO-COVAR method for K=10, L=3 and N=9. Monte-Carlo simulations with 100 trials.

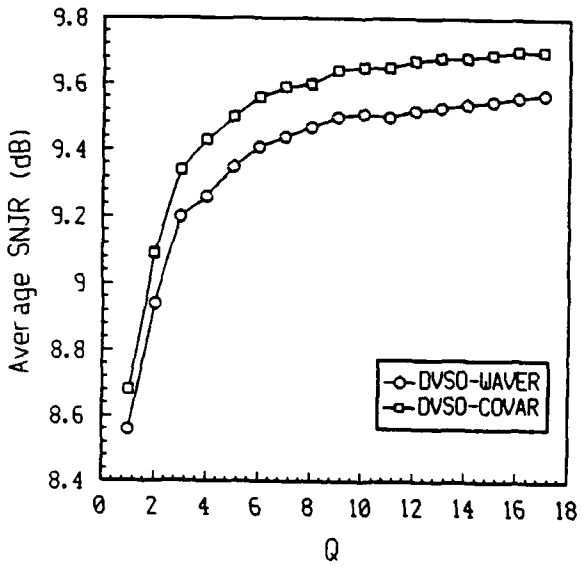

Fig 41 Average SNJR versus Q for DVSO-WAVER and DVSO-COVAR for K=10, L=1 and N=9. Monte-Carlo simulations with 100 trials.

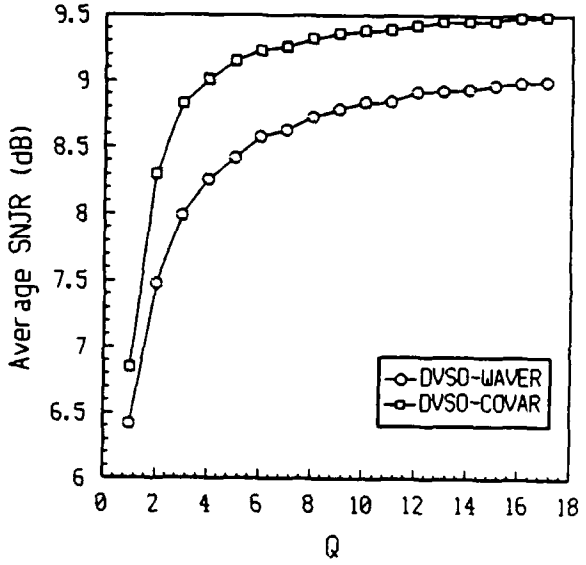

Fig 42 Average SNJR versus Q for DVSO-WAVER and DVSO-COVAR for K=10, L=3 and N=9. Monte-Carlo simulations with 100 trials.

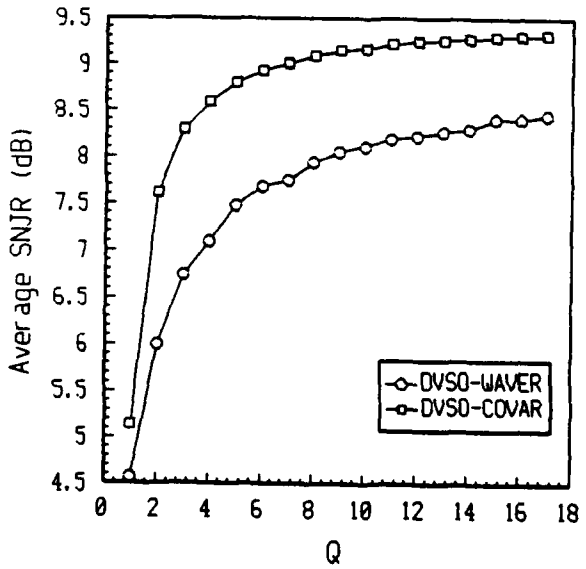

Fig 43 Average SNJR versus Q for DVSO-WAVER and DVSO-COVAR for K=10, L=5 and N=9. Monte-Carlo simulations with 100 trials.

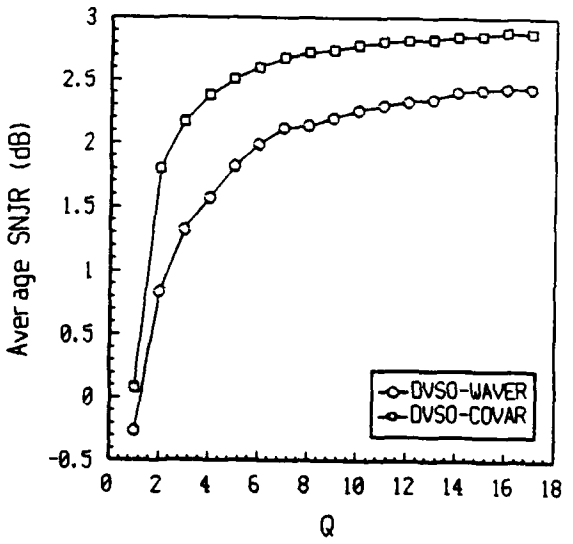

Fig 44 Average SNJR versus Q in a mainbeam jamming situation for DVSO-WAVER and DVSO-COVAR for K=10, L=3, N=9. Monte-Carlo simulations with 100 trials.

7.4 COMPARISON WITH THE SMI METHOD

Figures 45 to 47 give SNJR versus complex multiplications for DVSO–COVAR, DVSO–WAVER, DVSO and SMI methods for L=1, 3, 5 respectively (test case T_1). In Figure 45, when L<<K (in this case L/K=.1), the DVSO–WAVER method performs best for a SNJR below 9.3 dB and the DVSO–COVAR method performs best for a SNJR above 9.3 dB. For this example, the SMI method doesn't perform as well as the two others. The difference in performance between the DVSO–COVAR and the DVSO–WAVER method is not very big for small L/K ratios. In Figure 46, for L=3 (L/K=.3), we see that the three algorithms perform similarly even though DVSO–WAVER performs slightly better for SNJR below 8.1 dB and DVSO–COVAR performs slightly better for SNJR above 8.1 dB. Finally, in Figure 47, the four methods are compared for L=5 (L/K=.5). In this case, the SMI method performs best. We also note that the difference between DVSO–WAVER and DVSO–COVAR increases with L. For L>K/2 the DVSO–COVAR method still provides a high SNJR (at the expense of a high computation load); this is not the case for the basic DVO and DVSO.

We now consider the effect of using a larger array with K=40 elements. Figure 48 shows the SNJR versus complex multiplications for an array of K=40 elements with L=3 (using test case T_1). The DVSO method provides a SNJR about 3 dB below the SNJR optimum. We also see that the DVSO–COVAR method provides the same SNJR as SMI method but with 10 times fewer computations. This final example illustrates the principal area of advantage of the new techniques. They work best when the array is fairly large with a small number of jammers in comparison to the size of the array.

8.0 CONCLUSIONS AND SUMMARY

We have compared the performance of four fast projection methods, DVO, DVSO, DVSO–WAVER, DVSO–COVAR, with the performance of the sampled–matrix inversion (SMI) method.

We first showed that both the DVO and the DVSO methods use a data selection process but DVSO provides a better SNJR than DVO at the expense of a higher computational load. We also showed that since DVO and DVSO do not benefit from averaging techniques, they do not achieve as high a SNJR as does the SMI method.

We then presented two new algorithms which use a combination of data vector selection, Gram–Schmidt orthogonalization and averaging techniques to provide a high SNJR with a small computational load. It was shown that, when the number of jammers, L, is small with respect to the number of array elements, K, these two new algorithms are more efficient than SMI method i.e achieve the same SNJR but require far fewer computations. Moreover, it was shown that the DVSO–WAVER technique performs best for small SNJR but that DVSO–COVAR performs best for higher SNJR. Finally we showed that for both techniques, the SNJR approach a limiting asymptote as the number of multiplications increases – DVSO–COVAR approaches the optimum value while DVSO–WAVER has a slightly lower asymptote.

While this study did not consider non–stationary jamming, it is believed

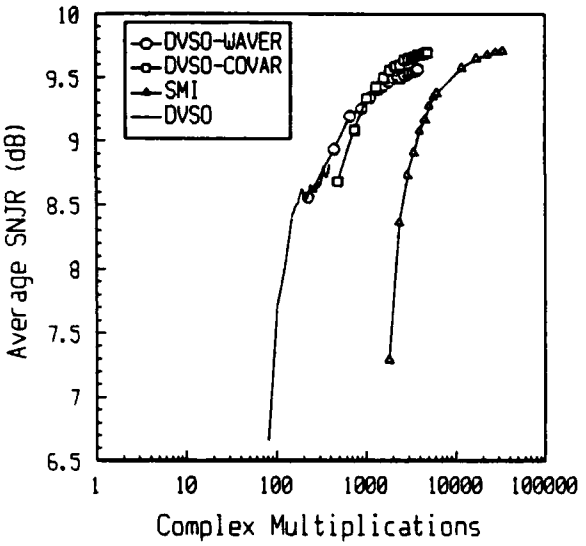

Fig 45 Average SNJR versus complex multiplications for the DVSO-WAVER, DVSO-COVAR, SMI and DVSO methods for K=10 and L=1. Monte-Carlo simulations with 100 trials.

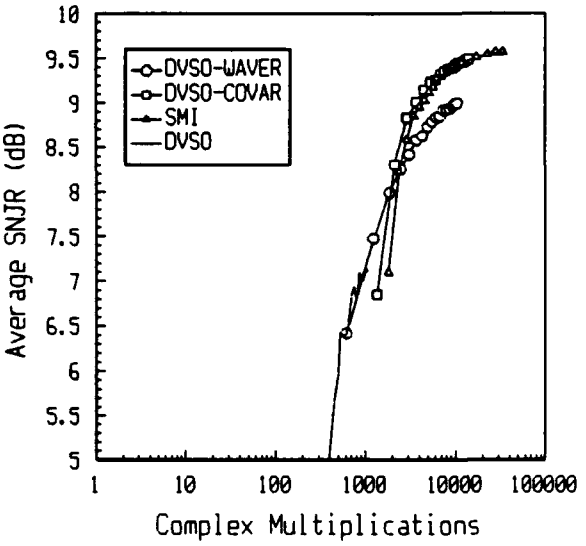

Fig 46 Average SNJR versus complex multiplications for the DVSO-WAVER, DVSO-COVAR, SMI and DVSO methods for K=10 and L=3. Monte-Carlo simulations with 100 trials.

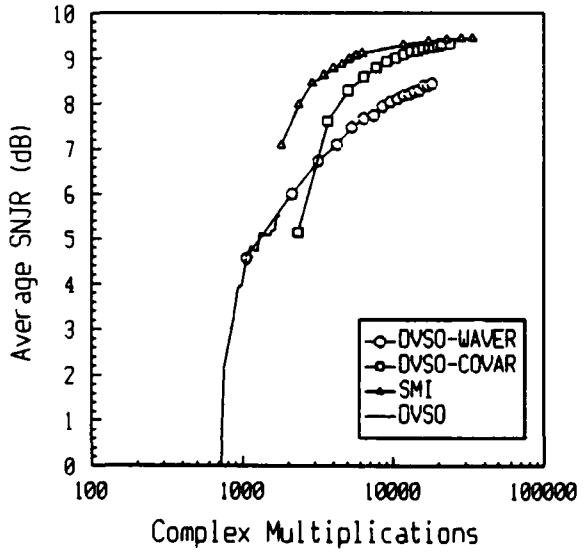

Fig 47 Average SNJR versus complex multiplications for the DVSO-WAVER, DVSO-COVAR, SMI and DVSO methods for K=10 and L=5. Monte-Carlo simulations with 100 trials.

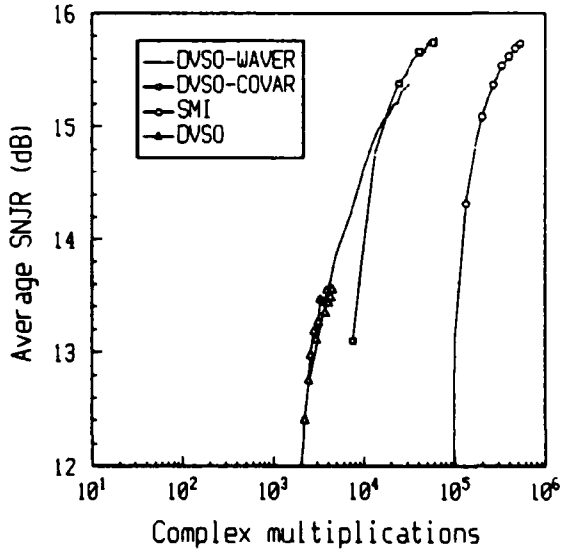

Fig. 48 Average SNJR versus complex multiplications for DVSO-WAVER, DVSO-COVAR, SMI and DVSO methods for for K=40 and L=3. Monte-Carlo simulations with 100 trials.

that DVSO–WAVER is likely to be more susceptible to blinking jamming than is DVSO–COVAR or DVSO itself.

Future work will consider the application of those techniques to angle–of–arrival estimation.

9.0 ACKNOWLEDGEMENTS

The authors thank Dr. H. C. Chan for reviewing the report and for his helpful suggestions.

10.0 REFERENCES

1. Reed, I.S., Mallett, J.D., Brennan, L.E., "Rapid Convergence Rate in Adaptive Arrays", IEEE Trans. Vol. AES–10, No.6, November 1974, pp.853–864.

2. Schmidt, R.O. "Multiple Emitter Location and Signal Parameter Estimation", IEEE Trans., Vol. AP–24, 1986, pp. 276–280.

3. Hung, E.K.L., Turner, R.M., "A Fast Beamforming Algorithm fir Large Arrays", IEEE Trans. Vol. AES–19, No–4, July 1983, pp.598–607.

4. Hung, E.K.L., Turner, R.M., Herring, R.W., "Fast and Effective Adaptive Beamforming and Spectral Line Estimation by Direct Orthogonalization of the Data Vectors", IEEE Asilomar Conference on Circuits, Systems and Computers, Oct. 31–Nov. 2, 1983, pp. 276–281.

5. Nickel, U., "Some Properties of Fast Projection Methods of the Hung–Turner Type", Signal Processing III: Theories and Applications, I.T Young et al (eds.), Signal Processing III: Theories and Application, North Holland, 1986, pp.1165–1168.

6. Toulgoat, M., Turner, R.M., "Sample Vectors Selection Applied to the Estimation of the Signal Subspace for Adaptive Nulling and Estimation of Direction of Arrival", DREO Signal Processing Workshop, Defence Research Establishment Ottawa, May 6–8, 1991.

7. Toulgoat, M., Turner, R.M., "The Application of Sample Vector Selection to Adaptive Nulling: Performance Comparison for Techniques Combining Gram–Schmidt Orthogonalization with Averaging", Canadian Conference on Electrical and Computer Engineering, Quebec, Sep. 25–27, 1991.

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1. ORIGINATOR (the name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Establishment sponsoring a contractor's report, or tasking agency, are entered in section 8.)

DEFENCE RESEARCH ESTABLISHMENT OTTAWA

2. SECURITY CLASSIFICATION (overall security classification of the document. including special warning terms if applicable)

UNCLASSIFIED

3. TITLE (the complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S,C or U) in parentheses after the title.)

Data Selection for Fast Projection Techniques Applied to Adaptive Nulling: A Comparative Study of Performance (U)

4. AUTHORS (Last name, first name, middle initial)

Toulgoat, Mylène, and Turner, Ross M.

5. DATE OF PUBLICATION (month and year of publication of document)	6a. NO. OF PAGES (total containing information. Include Annexes, Appendices, etc.)	6b. NO. OF REFS (total cited in document)
December 1991	36	7

7. DESCRIPTIVE NOTES (the category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)

Technical Report

8. SPONSORING ACTIVITY (the name of the department project office or laboratory sponsoring the research and development. Include the address.)

Defence Research Establishment Ottawa, 3701 Carling ave. Ottawa, Ontario, K1A 0Z4 Canada.

9a. PROJECT OR GRANT NO (if appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant)	9b. CONTRACT NO. (if appropriate, the applicable number under which the document was written)
041LC	

10a. ORIGINATOR'S DOCUMENT NUMBER (the official document number by which the document is identified by the originating activity. This number must be unique to this document.)	10b. OTHER DOCUMENT NOS. (Any other numbers which may be assigned this document either by the originator or by the sponsor)
DREO REPORT 1100	

11. DOCUMENT AVAILABILITY (any limitations on further dissemination of the document, other than those imposed by security classification)

(X) Unlimited distribution
( ) Distribution limited to defence departments and defence contractors; further distribution only as approved
( ) Distribution limited to defence departments and Canadian defence contractors; further distribution only as approved
( ) Distribution limited to government departments and agencies; further distribution only as approved
( ) Distribution limited to defence departments; further distribution only as approved
( ) Other (please specify):

12. DOCUMENT ANNOUNCEMENT (any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availability (11). However, where further distribution (beyond the audience specified in 11) is possible, a wider announcement audience may be selected.)

Unclassified

SECURITY CLASSIFICATION OF FORM

13. ABSTRACT (a brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), or (U). It is not necessary to include here abstracts in both offical languages unless the text is bilingual).

This report describes simulation studies of fast projection techniques for adaptive nulling of jammer signals. These techniques are based on a data selection criterion applied directly to the data vectors obtained at the output of an antenna array. The selected data vectors are used in fast projection algorithms to estimate the antenna weighting vector that is orthogonal to the jamming vectors. Four such algorithms are described ranging from the fastest but least effective to the most computationally demanding but most effective. In effect, these algorithms provide a better trade-off of performance versus computational load than has heretofore been available. Relative performance is compared to that of the standard Sample Matrix Inversion (SMI) technique for test cases which evaluate the effects of the number of jammers, the jammer strengths relative to the receiver noise and the jammer angular positions.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus, e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus-identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

ADAPTIVE NULLING
FAST PROJECTION TECHNIQUES
RADAR
PHASED ARRAY

UNCLASSIFIED

SECURITY CLASSIFICATION OF FORM